

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization International Bureau



(43) International Publication Date
6 May 2004 (06.05.2004)

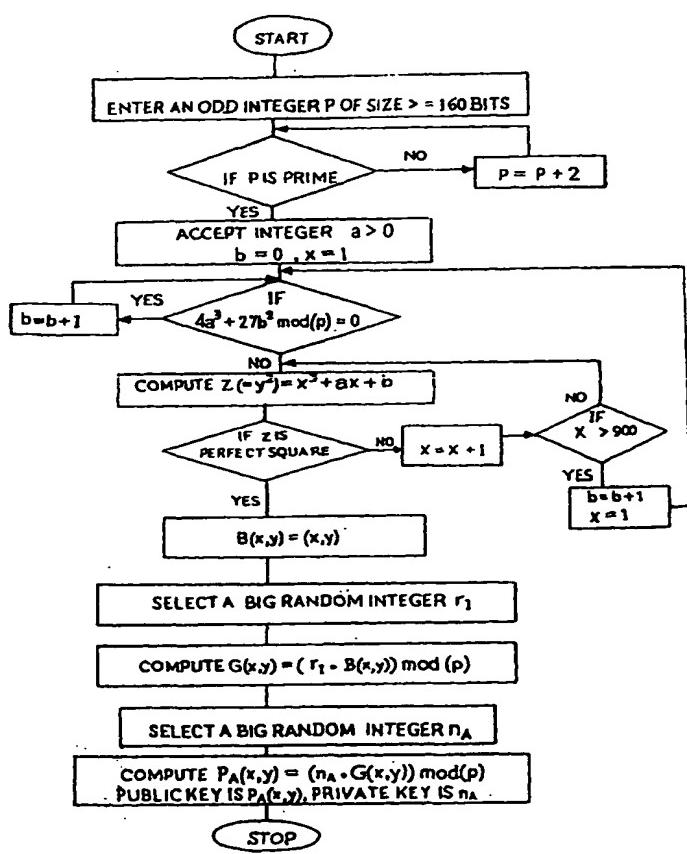
PCT

(10) International Publication Number
WO 2004/038680 A1

- (51) International Patent Classification⁷: **G09C 1/00**, (72) Inventors; and
H04L 9/30, 9/26 (75) Inventors/Applicants (*for US only*): **SRUNGARAM, Gopala, Krishna, Murthy [IN/IN]; Defence Research & Development, Laboratory, Kanchanbagh, Hyderabad 500 058 (IN). BISWAS, Rathindra, Nath [IN/IN]; E20/01, Lab Quarters, Kanchanbagh, Hyderabad-500 058 (IN).**
- (21) International Application Number: **PCT/IN2003/000339**
- (22) International Filing Date: 20 October 2003 (20.10.2003)
- (25) Filing Language: English (74) Agents: **DAVAR, G., S. et al.; L.S. Davar & Co., "Monalisa", Flats 1B & 1C, 17 Camac Street, Kolkata 700 017 (IN).**
- (26) Publication Language: English
- (30) Priority Data:
689/Del/02 26 October 2002 (26.10.2002) IN (81) Designated States (*national*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, OM, PH, PL, PT, RO, RU, SD, SE, SG, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZM, ZW.
- (71) Applicant (*for all designated States except US*): **THE ADDITIONAL DIRECTOR (IPR), DEFENCE RESEARCH & DEVELOPMENT ORGANISATION [IN/IN]; Ministry of Defence, Government of India, B-341, Sena Bhawan, DHQ PO, New Delhi 110 011, India (IN).**

[Continued on next page]

(54) Title: A METHOD OF ELLIPTIC CURVE ENCRYPTION



(57) Abstract: A method of elliptic curve encryption comprising the step of, (a) selecting an elliptic curve $E_p(a,b)$ of the form $y^2=x^3+ax+b \bmod p$ wherein a and b are non-negative integers less than p satisfying the formula $4a^3+27b^2 \bmod p$ not equal to 0; (b) generating a large 160 bit random number by a method of concatenation of a number of smaller random numbers; (c) generating a well hidden point $G(x,y)$ on the elliptic curve $E_p(a,b)$ by scalar multiplication of a point $B(x,y)$ on the elliptic curve with a large random integer which further comprises the steps; (i) converting the large random Integer Into a series of powers of 2^{31} ; (ii) converting each coefficient of 2^{31} obtained from above step into a binary series; (iii) multiplication of binary series obtained from steps(i) & (ii) above with the point $B(x,y)$ on the elliptic curve ; (d) generating a private key n_A (of about ≥ 160 bit length); (e) generating of public key $P_A(x,y)$ given by the formula $P_A(x,y) = (n_A \cdot G(x,y)) \bmod p$; (f) encrypting the input message MSG; (g) decrypting the ciphered text.



- (84) Designated States (*regional*): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

— before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

Published:

- with international search report

FIELD OF INVENTION

This invention relates to a method of elliptic curve encryption.

PRIOR ART

Data security, authentication and verification are desirable features in
5 the Internet based data communications, wireless communication, E-Commerce and smart card related applications etc. Basically, data encryption systems can be divided into two categories: symmetric encryption systems and asymmetric encryption systems. In a symmetric encryption system, the same key is used to encrypt the data at sender's end and to decrypt the ciphered text at the receiver's end.
10 However, in such systems, the encryption key is required to be exchanged beforehand over a secure communication channel.

Asymmetric encryption systems utilize two separate keys for encryption of the data and decryption of the ciphered text. The key to encrypt the data is made public while the corresponding decryption key is kept private and not shared with other. The private key can not be generated from the public key and, as such, only the intended recipient with the private key can decrypt the ciphered text. Asymmetric encryption systems do not need the prior exchange of keys and hence
15 are preferred over symmetric encryption systems. The most well known asymmetric encryption system is RSA encryption system. The RSA encryption system is based on integer factorization problem.
20

In RSA algorithm, two primes p and q, usually very large, are required to generate a modulus n, with the equation $n = p \cdot q$. In RSA algorithm the public key d and private key e are related with the equation

$$e \cdot d = 1 \pmod{\theta} \quad (\text{sign stands for multiplication})$$

5 Where, $\theta = (p-1) \cdot (q-1)$

The input message M is encrypted with the equation

$$M_c = (M)^d \pmod{n}$$

Where M_c is cipher of the input message M, d is the public key and n is the modulus. M_c can be reconstructed to the input message M with the
10 equation

$$M = (M_c)^e \pmod{n}$$

In RSA algorithm, the private and public keys are chosen sufficiently big to achieve an adequate level of security. The security of the system is based on the principle of difficulty in factoring a large number that has no relatively small factors. Accordingly, p and q must be relatively large prime numbers. As the advances already made in crypt analysis system and computation speed are threats to the encryption systems utilizing moderate sized keys, bigger and bigger sized keys are being used for encryption systems. Ultimately, the size of the key (n) is
15 required to be around 1024 bits to achieve an adequate level of security. Due to the need of bigger key size and nature of operation, RSA algorithm demands more memory, bandwidth for communication and computation time.
20

25 However, the RSA technique, already known in the art, suffers from the following disadvantages.

Main disadvantage of the RSA encryption system, known in the art, is that it requires significant band width and storage capacity.

Another disadvantage of the RSA encryption technique, known in the art, is that it requires more time for computation and communication

- 5 Yet another disadvantage of the RSA encryption system, known in the art, is that it is vulnerable particularly in view of the recent advances in the analytical techniques.

- An alternate encryption system, Digital Signature Algorithm (DSA) is based on discrete logarithm problem on finite group. This encryption 10 system is widely used for digital signature for authentication.

- If G is a finite group and a and b are elements of G , then the equation $a^x = b$ represents a discrete logarithm problem. If a and x are known, finding b is straight forward. Here the value x is called logarithm of b to the base a , i.e. $x = \log_a b$. Finding the value of x is more difficult, if a and b are sufficiently large.

- A variation of discrete logarithm problem is the elliptic curve discrete logarithm problem. In this case, the discrete logarithm is based on an elliptic curve $E_p(a,b)$, defined on a finite field. It is well known that solving a problem based on elliptic curve discrete logarithm is more 20 difficult than a problem based on discrete logarithm based on finite group. In the elliptic curve cryptography method, each person can define his own elliptic curve for encryption and decryption, thus providing increased security. An elliptic curve can be easily redefined and new public and private keys can be generated to return to a 25 secure system. The elliptic curve method reduces the bandwidth requirement of the public key system because the parameters can be

stored with fewer keys. This is an important feature, which helps in restricting the key size in elliptic curve cryptography. The elliptic curve method of encryption is well known in the art. However, the elliptic curve methods, known in the art, suffer from following disadvantages.

- 5 Main disadvantage of the elliptic curve method of encryption, known in the art, is that the scalar multiplication involved in the encryption process takes large computer time thereby rendering the entire encryption slower and unsuitable for applications where time factor is very critical.
- 10 Another disadvantage of the elliptic curve method, known in the art, is that the encryption process utilises only one coordinate of a point on the elliptic curve for encoding the message thereby reducing the throughput of the encryption system.

OBJECTS OF THE INVENTION

- 15 Primary object of the invention is to provide a method of elliptic curve encryption, which is based on discrete logarithm problem on elliptic curve.

- Another object of the invention is to provide a method of elliptic curve encryption, which has a higher throughput as long streams of messages can be encrypted with same set of points.
- 20

Yet another object of the invention is to provide a method of elliptic curve encryption, which uses an efficient method of multiplication of a point on the elliptic curve $E_p(a, b)$ by a large integer thereby reducing the encryption time.

Still another object of the invention is to provide a method of elliptic curve encryption, which has higher security level as it selects different random points lying on the curve for different points of the message.

Yet further object of the invention is to provide a method of elliptic curve encryption, which utilises both x and y coordinates of points on the curve corresponding to a point in the plane generated from message thereby enhancing the throughput.

Still another object of the invention is to provide a method of elliptic curve encryption, which provides a separate method of generating random numbers thereby facilitating the realization of higher level of security.

Yet further object of the invention is to provide an improved elliptic curve encryption system, which provides an efficient binary series representation of big integer thereby optimising the scalar multiplication time by reducing number of operations.

SUMMARY OF THE INVENTION

According to the present invention, there is provided a method of elliptic curve encryption based on elliptic curve method. The inherent security provided by the elliptic curve is derived from the characteristic that the addition of two points on the curve can be defined as another point on the curve. If a starting point is selected on the curve and is multiplied by an integer, the new point also lies on the elliptic curve. The present elliptic curve encryption method has lower bandwidth requirement and has reduced encryption time. The encryption method has enhanced security as it selects different random points lying on the curve corresponding to different points of the input message. The

method provides an efficient method for selection of random points on the elliptic curve. The encryption method of the present invention utilizes x and y coordinates, both, corresponding to the message thereby increasing the throughput of the system. The present invention also provides an efficient method to convert a big integer into a series of powers of 2, which reduces division and multiplication operations. It also provides an efficient method of scalar multiplication of a point on the elliptic curve by a large integer thereby reducing the encryption time.

10 **DESCRIPTION OF THE DRAWINGS**

- Fig. 1 is a list of steps involved in the generation of large random integer
- Fig. 2 is a list of steps to convert a large integer into a series of numbers which are multiples of $(2^{31})^n$, where each number is less than 2^{31} .
- Fig. 3 is a list of steps to convert each of the coefficient of the series of numbers, which are multiples of $(2^{31})^n$, into a binary series.
- Fig. 4 is a list of steps involved in multiplication of a binary series with a point on the elliptic curve.
- 20 Fig. 5 is a list of steps involved in the generation of encryption keys
- Fig. 6 is a list of steps involved in the encryption of input message
- Fig. 7 is a list of steps involved in the decryption of the encrypted message

25 **DESCRIPTION OF THE INVENTION**

Any encryption system based on elliptic curve cryptography derives its strength from elliptic curve discrete logarithm problem. An elliptic curve $E_p(a,b)$ over a finite field has, in general, the following form

$$Y^2 = X^3 + aX + b \text{ mod } (p)$$

Where, p is a large prime number and a, b are non-negative integers less than p that satisfy the following equation.

$$4a^3 + 27b^2 \text{ mod}(p) \text{ not equal to } 0$$

- 5 In this algorithm, we have taken p as a 160-bit length (approximately 49 decimal digits) prime number. In this equation, selection of a, b & p decides the elliptic curve.

10 The purpose of secrecy is served, if a well hidden point G(x,y) on the elliptic curve is selected. This condition can be satisfied if coordinates x and y are large enough to be unpredictable. Finding such a well-hidden point on the elliptic curve is a challenging task. To solve this problem, the present method of encryption utilises the well-known property of the elliptic curve, that a scalar multiplication of a point on the elliptic curve gives another point on the elliptic curve. In the present method of encryption, initially a point on the curve is selected by scanning a limited range of x and then this value is multiplied by a large random integer to realize the required well hidden point on the elliptic curve.

15 In the present method of encryption, a large (160 bit) random integer r_1 is used to choose a point G(x,y) on the elliptic curve $E_p(a,b)$, where x and y coordinate values are also large. The random number r_1 is generated by a method of concatenation of a number of smaller random numbers. Once the point G(x,y) is known, the private key n_A (approx 160 bits length) can be selected manually or by any predefined method. For the purpose of automation, a random integer n_A has been considered as a private key. Then public key $P_A(x,y)$ is given by the formula

$$P_A(x,y) = n_A \cdot G(x,y) \text{ mod}(p)$$

Where stands for multiplication of point $G(x,y)$ on an elliptic curve with a random integer n_A . Here $P_A(x,y)$ is also a point on the elliptic curve $E_p(a,b)$. Here both co-ordinates x and y are large and as such it is very difficult to predict or calculate n_A , even if the equation of the curve and public key information are made available. The improved elliptic curve encryption system, of the present invention, can be described in following steps with the help of corresponding figures.

- (I) Generating an elliptic curve**
- An equation of elliptic curve $E_p(a,b)$ is generated by selecting two integers a & b which satisfy the following equation.

$$y^2 = x^3 + ax + b \text{ mod } (p)$$

Where, $4a^3 + 27b^2 \text{ mod } (p)$ not equal to 0

- The elliptic curve equation is generated, while generating encryption keys, as described in step (IV)

- (II) Generating a large random integer**
- It is extremely important to generate a random point $G(x,y)$ on the elliptic curve which has a very large value for its coordinates (of the order of 160 bit) to ensure secure encryption. In order to realize this, it is essential to generate large random integers. Selecting number of small random integers (less than 10 digits) and concatenating these random integers generates the large random integer. Large random numbers are also used elsewhere in the algorithm for the purpose of key generation and masking.

Referring to Fig. 1, the generation of a large random Integer (say M) comprises of following steps:

- 5 (i) setting I = 0
- (ii) setting M to null
- (iii) determining whether I < 6
- (iv) going to next if true
- (v) returning M as result if false
- (vi) generating a random number RI within (0,1) by using library function
- 10 (vii) multiplying RI with 10^8 to obtain BINT-an integer of size 9 digits
- (viii) concatenating BINT to M
- (ix) setting I = I+1
- (x) returning to step(iii)

The above procedure generate a big random Integer M with the size of approximately 160 bits (49 decimal digits approximately 160 bits).

15 (III) Generating a well hidden point on the elliptic curve by Scalar Multiplication of a large random Integer with a point on elliptic curve

20 Scalar multiplication of a point B (x,y) on the elliptic curve with a large random Integer (say r_1) generates a well hidden point G (x,y) on the elliptic curve due to a well known property of the elliptic curve. A random point B (x,y) on the elliptic curve $E_p(a,b)$ is arbitrarily obtained by scanning a limited range of values [1,900] for x on the elliptic curve.

25 In the present invention, a new algorithm for performing scalar multiplication has been proposed. The process of scalar multiplication of the present invention optimises the computational time for performing the scalar multiplication. The scalar multiplication process is required for generation of well hidden point on the elliptic curve as well for generation of encryption keys, generation of ciphered text and deciphering of ciphered text. Scalar multiplication of a point on the elliptic curve with any large integer can be performed by repeated

addition of the point on the elliptic curve. This optimised multiplication procedure requires binary series for addition of points which, in turn, demands representation of a large integer in powers of 2. This is achieved in following three steps.

5 (a) Conversion of the large random Integer into a binary series

The random integer (M) is converted into a binary series of following type

$$M = m_0 (2^{31})^0 + m_1 (2^{31})^1 + \dots + m_n (2^{31})^n$$

$$\text{Where each } m_n (< 2^{31}) = c_0 2^0 + c_1 2^1 + \dots + c_{30} 2^{30}$$

10 And c_0, c_1, \dots, c_{30} are zero or one

Now, the scalar multiplication of 2 with $B(x,y)$ can also be considered as addition of $B(x,y)$ and $B(x,y)$.

$$2. B(x,y) = B(x,y) + B(x,y)$$

and similarly,

15 $2^2 \cdot B(x,y) = 2^1 \cdot B(x,y) + 2^1 \cdot B(x,y)$ and so on

$$2^n \cdot B(x,y) = 2^{(n-1)} \cdot B(x,y) + 2^{(n-1)} \cdot B(x,y) \text{ and so on}$$

(b) Addition of two points on the elliptic curve

This addition is achieved by using the following formula

$$B_3(x,y) = B_1(x,y) + B_2(x,y) \text{ where}$$

20 X coordinate of $B_3(x,y) = s^2 - B_1(x) - B_2(x) \bmod (p)$

And

$$Y \text{ coordinate of } B_3(x,y) = s(B_1(x) - B_3(x) - B_1(y)) \bmod (p)$$

$$\text{Where, } s = (B_2(y) - B_1(y)) / (B_2(x) - B_1(x))$$

$$\{\text{if } B_1(x,y) = B_2(x,y)\}$$

25 $s = (3B_1^2(x) + a) / 2B_1(y)\}$

Referring to Fig. 2, Fig.3 & Fig.4, the scalar multiplication of a random integer with a point on the elliptic curve comprises of following steps:

(a) Converting big integer into a series of powers of 2^{31}

In the first step, the big integer M is divided with a value of 2^{31} to obtain a series of values $m_0, m_1, m_2 \dots m_n$ where the value of m_n lies in $[0, 2^{31})$ so that

$$5 \quad M = m_0 (2^{31})^0 + m_1 (2^{31})^1 + \dots + m_n (2^{31})^n$$

Referring to Fig. 2 & Fig.3 this comprises of following steps

- (I) accepting a big Integer M
- (II) setting T31 equal to 2^{31}
- (III) setting LIM =size of M (in bits) and initialize array A() with size LIM
- 10 (IV) setting INCRE equal to zero
- (V) setting N equal to M modulus T31
- (VI) setting M = INT (M/T31)
- (VII) determining whether N is equal to 0
- 15 (VIII) going to next if true
- (IX) going to step (xxiv) if false
- (X) determining whether M is equal to 0
- (XI) going to next if true
- (XII) going to step (xxvi) if false
- 20 (XIII) setting I & J equal to 0
- (XIV) determining whether I ≥ LIM
- (XV) going to next if false
- (XVI) going to step (xxviii) if true
- (XVII) determining whether A (I) is equal to 1
- 25 (XVIII) going to next if true
- (XIX) returning to step (xxii) if false
- (XX) setting B (J) equal to I
- (XXI) setting J = J+1
- (XXII) setting I = I+1
- 30 (XXIII) returning to step (xiv)
- (XXIV) calling function BSERIES (N, INCRE), which updates array A()
- (XXV) returning to step (x)
- (XXVI) setting INCRE = INCRE + 31
- (XXVII) returning to step (v)
- 35 (XXVIII) returning array B () as result

(b) Converting each coefficient m_n of the 2^{31} series obtained from above step further into a binary series

In this step, coefficients of the individual numbers in the 2^{31} series, obtained from above step, are converted into a series of powers of 2. A function BSERIES is used to convert each coefficient (m_n) into a series of powers of 2.

5 Referring to Fig. 2 and Fig.3, this comprises of following steps:

- (i) accepting N and INCRE from step (a)
- (ii) assigning BARRAY as an array of values which are powers of 2 ($[2^0, \dots, 2^{30}]$)
- 10 (iii) setting SIZE = size of N (in digits)
- (iv) computing POINTER = 3 (SIZE) +INT (SIZE/3)-4
- (v) determining whether POINTER <2
- (vi) going to next if true
- (vii) going to step (ix) if false
- (viii) setting POINTER equal to zero
- 15 (ix) determining whether (BARRAY(POINTER)≥N)
- (x) going to next if true
- (xi) going to step (xii) if false
- (xii) determining whether BARRAY (POINTER)=N
- (xiii) going to next if true
- 20 (xiv) going to step (xvii) if false
- (xv) setting A (POINTER + INCRE) equal to 1
- (xvi) returning array A () as result
- (xvii) setting A ((POINTER - 1)+ INCRE) equal to 1
- (xviii) computing N =N-BARRAY (POINTER-1)
- 25 (xix) returning to step (iii)
- (xx) setting POINTER = POINTER + 1
- (xxi) returning to step (ix)

(c) Multiplication of binary series obtained from steps (a) and (b) above with a point on the elliptic curve.

30 Referring to Fig. 2 & Fig. 4, the multiplication of binary series with a point on the elliptic curve comprises of following steps:

- (I) accepting $B(x,y)$, a point on $E_p(a,b)$
- (II) accepting array $B()$ with size LIM
- (III) setting I & J equal to zero
- (IV) determining whether $B(J)=I$
- 5 (V) going to next if true
- (VI) going to step (xxv) if false
- (VII) setting PARR(x,y) (J) equal to $B(x,y)$
- (VIII) setting $J=J+1$
- (IX) determining whether J is equal to LIM
- 10 (X) going to next if true
- (XI) going to step (xxv) if false
- (XII) setting K equal to zero
- (XIII) determining whether $K>0$
- (XIV) going to next if true
- 15 (XV) going to step (xxii) if false
- (XVI) computing $FP(x,y)=FP(x,y) + PARR(x,y)$ (K)
- (XVII) setting $K=K+1$
- (XVIII) determining whether $K=LIM$
- (XIX) going to next if true
- 20 (XX) returning to step (xiii) if false
- (XXI) returning $FP(x,y)$ as result
- (XXII) setting $FP(x,y)$ equal to $PARR(x,y)$ (K)
- (XXIII) setting $K=K+1$
- (XXIV) returning to step (xiii)
- 25 (XXV) setting $I=I+1$
- (XXVI) setting $B(x,y)=B(x,y) + B(x,y)$
- (XXVII) returning to step (IV)

(IV) Generating encryption keys

In order to create a public key, based upon the property of elliptic curve, discrete logarithm problem has to be established. For this, an arbitrary point on the elliptic curve, $B(x,y)$ is selected. Next, a random integer r_1 is generated by adopting the procedure, as described in step (II). Scalar multiplication of this point on the elliptic curve with the random integer is performed to generate a well hidden point $G(x,y)$ on the elliptic curve $E_p(a,b)$.

$$G(x,y) = r_1 \cdot B(x,y) \bmod (p)$$

The operation of scalar multiplication of random integer with the point on the elliptic curve is performed by adopting the procedure as described in step (III). Once the well hidden point $G(x,y)$ is known, the private key n_A (approximate 160 bit length) can be selected manually or by any predefined method. For the purpose of automation, a random number, n_A is considered as private key. The public key $P_A(x,y)$ is given by the formula.

$$P_A(x,y) = n_A \cdot G(x,y) \text{ mod}(p)$$

Once, the public key and the corresponding private key are determined, the input message can be encrypted and decrypted with these keys.

Referring to Fig. 5, the steps involved in generation of encryption keys are provided in the following.

- (i) entering a big odd integer p of size ≥ 160 bits
- 15 (ii) determining whether p is a prime number
- (iii) going to next if p is prime
- (iv) going to step (ix) if p is not prime
- (v) entering a small integer $a > 0$
- (vi) setting integer $b = 0$ & $x=1$
- 20 (vii) determining whether $4a^3+27b^2 \text{ mod}(p)$ is equal to zero
- (viii) going to next if false
- (ix) setting $b = b+1$ if true and going to step(vii)
- (x) setting $z (=y^2)$ equal to $x^3 + ax + b$
- (xi) determining whether z is a perfect square
- 25 (xii) going to step(xx) if z is not a perfect square
- (xiii) setting $B(x,y)=(x,y)$ if z is a perfect square
- (xiv) selecting a large random integer r_1
- (xv) setting $G(x,y)$ equal to $(r_1 \cdot B(x,y)) \text{ mod}(p)$
- (xvi) selecting a large random integer n_A
- 30 (xvii) setting $P_A(x,y)$ equal to $(n_A \cdot G(x,y)) \text{ mod}(p)$
- (xviii) return $P_A(x,y)$ as public key and n_A as private key
- (xix) setting $p=p+2$
- (xx) returning to step (ii)
- (xxi) setting $x=x+1$

- (xxii) determining whether $x > 900$
 (xxiii) going to next if true
 (xxiv) returning to step (x) if false
 (xxv) setting $b = b + 1$
 (xxvi) setting $x = 1$
 (xxvii) returning to step (vii)

(V) Encrypting the Input message

Since the message (say MSG) is in an alphanumeric form, it is necessary to convert this message in a collection of numbers. Taking corresponding ASCII value of each character of the input message creates these numbers. These numbers are linearised by adding 1000 to each of the ASCII value. Out of these bunch of numbers corresponding to ASCII equivalent, only 48 digits are selected at a time. Now, out of these sets of 48 digits, adjacent two numbers $P_c(x,y)$ are selected as a set of points. However, these points may not lie on the elliptic curve. It is essential that all the points, which are to be encrypted, must lie on the elliptic curve. In order to realize this, following procedure is adopted.

Points $P_{mask}(x,y)$ and $P_k(x,y)$ on the elliptic curve are generated by using the following formula

$$P_{mask}(x,y) = (K \cdot P_A(x,y)) \bmod(p)$$

$$P_k(x,y) = (K \cdot G(x,y)) \bmod p$$

Where K is a large random integer generated by following the procedure as described in step (II) above. Here $P_A(x,y)$ is the public key generated above and $G(x,y)$ is the well hidden point generated above. Similarly, another point $P_m(x,y)$ on the elliptic curve is generated with the help of following formula.

$$P_m(x,y) = (r_2 \cdot G(x,y)) \bmod(p)$$

Here r_2 is another random integer generated by using the procedure as described in step (II) above.

- 5 This point $P_m(x,y)$ is masked with the help of the point $P_{mask}(x,y)$ on the elliptic curve generated above.

$$P_{mk}(x,y) = (P_m(x,y) + P_{mask}(x,y)) \bmod(p)$$

The encrypted message $P_e(x,y)$ is generated from the following

$$P_e(x,y) = P_c(x,y) - P_m(x,y) \quad (\text{Here } -\text{stands for difference of Coordinates } x \text{ and } y \text{ of } P_c(x,y) \text{ and } P_m(x,y))$$

- 10 This process is repeated by selecting different random numbers for different set of (48,48) digits corresponding to the input message. This particular feature of the present encryption system enhances its security level. It is clear, from above, that from $P_e(x,y)$ and n_A the original message can not be reconstructed. In order to decrypt the ciphered message, it is essential to transmit $P_e(x,y)$, $P_{mk}(x,y)$ and $P_k(x,y)$. However, since $P_{mk}(x,y)$ and $P_k(x,y)$ are points on the elliptic curve $E_p(a,b)$, only x coordinate of these points need to be transmitted. Y coordinates of these points can be computed at other end by using elliptic curve $E_p(a, b)$. $P_e(x,y)$ is the message hidden with the help of a third point $P_m(x,y)$ on the elliptic curve $E_p(a,b)$.

- 20 Referring to Fig 6, the encryption process comprises of following steps:

- (I) generating a large random Integer K
- (II) setting $P_{mask}(x,y) = k \cdot P_A(x,y) \bmod (p)$
- (III) setting $P_k(x,y) = k \cdot G(x,y) \bmod (p)$
- 25 (IV) accepting the message (to be encrypted)

- (v) converting the message into a point $P_c(x,y)$
- (vi) generating a random point $P_m(x,y)$ on elliptic curve $E_p(a,b)$
- (vii) setting $P_e(x,y) = (P_c(x,y) - P_m(x,y))$ (here-stands for difference of coordinates)
- 5 (viii) setting $P_{mk}(x,y) = (P_m(x,y) + P_{mask}(x,y)) \bmod(p)$
- (ix) returning $P_k(x)$, $P_e(x,y)$ and $P_{mk}(X)$ as the result (cipher)

(VI) Decrypting the encrypted message

Referring to Fig. 7, the decryption process reconstructs the message (MSG) from the ciphered message by using the following formula.

$$\begin{aligned}
 10 \quad P_c(x,y) &= P_e(x,y) + P_m(x,y) \\
 &= P_e(x,y) + (P_{mk}(x,y) - k \cdot P_A(x,y)) \\
 &= P_e(x,y) + (P_{mk}(x,y) - k \cdot (n_A \cdot G(x,y))) \\
 &= P_e(x,y) + (P_{mk}(x,y) - n_A \cdot (k \cdot G(x,y))) \\
 &= P_e(x,y) + (P_{mk}(x,y) - n_A \cdot P_k(x,y))
 \end{aligned}$$

15 Here, $P_e(x,y)$, $P_{mk}(x,y)$ and $P_k(x,y)$ are obtained from transmitted values and n_A is private key and these values are sufficient to reconstruct the message.

Referring to Fig. 7, The decryption of the ciphered message comprises of following steps:

- 20 (i) getting cipher text ($P_k(x)$, $P_e(x,y)$ and $P_{mk}(X)$)
- (ii) computing $P_k(y)$ from $P_k(x)$ by using elliptic curve $E_p(a,b)$
- (iii) computing $P_{mk}(y)$ from $P_{mk}(x)$ by using elliptic curve $E_p(a,b)$
- (iv) computing $P_{ek}(x,y) = (n_A \cdot P_k(x,y)) \bmod(p)$

- (v) computing $P_m(x,y) = (P_{mk}(x,y) - P_{ek}(x,y)) \text{ mod}(p)$
- (vi) computing $P_c(x,y) = P_m(x,y) + P_e(x,y)$ (here + stands for addition of coordinates)
- (vii) converting $P_c(x,y)$ into the input message MSG

It is to be understood that the process of the present invention is
5 susceptible to adaptations, changes and modifications by those skilled in
the art. Such adaptations, changes and modifications are intended to be
within the scope of the present invention, which is further set forth with
the following claims.

WE CLAIM

1. A method of elliptic curve encryption comprising the step of:
 - (a) selecting an elliptic curve $E_p(a,b)$ of the form $y^2=x^3+ax+b \text{ mod } (p)$ wherein a and b are non-negative integers less than p satisfying the formula $4a^3 + 27b^2 \text{ mod } (p)$ not equal to 0;
 - 5 (b) generating a large 160 bit random number by a method of concatenation of a number of smaller random numbers;
 - (c) generating a well hidden point $G(x,y)$ on the elliptic curve $E_p(a,b)$ by scalar multiplication of a point $B(x,y)$ on the elliptic curve with a large random integer which further comprises the steps;
 - 10 (d) converting the large random integer into a series of powers of 2^{31} ;
 - (e) converting each coefficient of 2^{31} obtained from above step into a binary series;
 - (f) multiplication of binary series obtained from steps(d) & (e) above with the point $B(x,y)$ on the elliptic curve
 - 15 (g) generating a private key n_A (of about $>= 160$ bit length);
 - (h) generating of public key $P_A(x,y)$ given by the formula $P_A(x,y) = (n_A \cdot G(x,y)) \text{ mod } (p)$;
 - (i) encrypting the input message MSG;
 - (j) decrypting the ciphered text
- 20 2. A method of elliptic curve encryption as claimed in claim 1, wherein the said number p appearing in selection of elliptic curve is about 160 bit length prime number.
 3. A method of elliptic curve encryption as claimed in claim (1), wherein the said method of generating any large random integer M comprises the steps of;

- (i) setting $I = 0$;
 - (ii) setting M to null;
 - (iii) determining whether $I < 6$;
 - (iv) going to next if true;
 - 5 (v) returning M as result if false;
 - (vi) generating a random number R_I within $(0,1)$ by using library function;
 - (vii) multiplying R_I with 10^9 to obtain BINT – an integer of size 9 digits;
 - (viii) concatenating BINT to M ;
 - 10 (ix) setting $I = I + 1$;
 - (x) returning to step (iii)
4. A method of elliptic curve encryption as claimed in claims 1 to 3 wherein the said conversion of large random integer into a series of powers of 2^{31} and said conversion of each coefficient m_n of the said 2^{31} series thus obtained for scalar multiplication for the said random integer with the said point $B(x,y)$ on the said elliptic curve $E_p(a,b)$ comprises the steps of:
- (i) accepting a big integer M ;
 - (ii) setting $T31$ equal to 2^{31} .
 - 20 (iii) setting LIM = size of M (in bits) and initializing array $A()$ with size LIM ;
 - (iv) setting $INCRE$ equal to zero;
 - (v) setting N equal to M modulus $T31$;
 - (vi) setting $M = INT(M/T31)$;
 - 25 (vii) determining whether N is equal to 0;
 - (viii) going to next if true;
 - (ix) going to step (xxiv) if false;
 - (x) determining whether M is equal to 0;
 - (xi) going to next if true;

- (xii) going to step (xxvi) if false;
 - (xiii) setting I =0 & J=0;
 - (xiv) determining whether I ≥ LIM;
 - (xv) going to next step if false;
 - 5 (xvi) going to step (xxviii) if true;
 - (xvii) determining whether A(I) is equal to 1;
 - (xviii) going to next step if true;
 - (xix) returning to step (xxii) if false;
 - (xx) setting B (J) =I;
 - 10 (xxi) incrementing J by 1;
 - (xxii) Incrementing I by 1;
 - (xxiii) returning to step (xiv);
 - (xxiv) calling function BSERIES (N, INCRE) and updating array A ();
 - (xxv) returning to step (x)
 - 15 (xxvi) setting INCRE = INCRE + 31;
 - (xxvii) returning to step (v);
 - (xxviii) returning array B () as result.
5. A method of elliptic curve encryption as claimed in claims 1 to 4, wherein the said conversion of large random integer into a series of powers of 2^{31} and said conversion of each coefficient m_n of the said 2^{31} series thus obtained for the said scalar multiplication of the said random integer with the said point $B(x,y)$ on the said elliptic curve $E_p(a,b)$ further comprises the steps of:
- (i) accepting N and INCRE;
 - 25 (ii) assigning BARRAY as an array of values which are powers of $2([2^0, \dots, 2^{30}])$;
 - (iii) setting SIZE =size of N (in digits);
 - (iv) computing POINTER = 3 (SIZE)+INT(SIZE/3)-4;
 - (v) determining whether POINTER < 2;
 - 30 (vi) going to next if true;

- (vii) going to step (bx) if false;
 - (viii) setting POINTER equal to zero;
 - (ix) determining whether (BARRAY(POINTER) ≥N);
 - (x) going to next step if true;
 - 5 (xi) going to step (xx) if false;
 - (xii) determining whether BARRAY (POINTER)=N;
 - (xiii) going to next step if true;
 - (xiv) going to step (xvii) if false;
 - (xv) setting A (POINTER + INCRE) equal to 1;
 - 10 (xvi) returning array A () as result;
 - (xvii) setting A ((POINTER - 1) + INCRE) equal to 1;
 - (xviii) computing N=N-BARRAY(POINTER-1);
 - (xix) returning to step (iii);
 - (xx) setting POINTER = POINTER + 1;
 - 15 (xxi) returning to step (ix);
6. A method of elliptic curve encryption as claimed in claims 1 to 5, wherein the said scalar multiplication of the said binary series with the said point B(x,y) on the said elliptic curve $E_p(a,b)$ comprises the steps of:
- 20 (I) accepting B(x,y), a point on $E_p(a,b)$;
 - (II) accepting array B() with size LIM;
 - (III) setting I = 0 & J=0;
 - (IV) determining whether B(J)=I;
 - (V) going to next step if true;
 - 25 (VI) going to step (xxv) if false;
 - (VII) setting PARR (x,y) (J) equal to B(x,y);
 - (VIII) incrementing J by 1;
 - (IX) determining whether J is equal to LIM;
 - (X) going to next step if true;

- (xi) going to step (xxv) if false;
- (xii) setting K=zero;
- (xiii) determining whether K>0;
- (xiv) going to next step if true;
- 5 (xv) going to step (xii) if false;
- (xvi) computing $FP(x,y)=FP(x,y) + PARR(x,y) (K)$;
- (xvii) incrementing K by 1;
- (xviii) determining whether K=LIM;
- (xix) going to next if true;
- 10 (xx) returning to step (xiii) if false;
- (xxi) returning FP(x,y) as result;
- (xxii) setting FP(x,y) equal to PARR(x,y) (K);
- (xxiii) incrementing K by 1;
- (xxiv) returning to step (xiii);
- 15 (xxv) incrementing I by 1;
- (xxvi) setting $B(x,y) = B(x,y) + B(x,y)$;
- (xxvii) returning to step (iv)..

- 7. A method of elliptic curve encryption as claimed in claim 1, wherein the said public key $P_A(x,y)$ is also a point on the said elliptic curve $E_p(a,b)$.
- 20 8. A method of elliptic curve encryption as claimed in claims 1 to 7, wherein the said generation of the said private key n_A and the said public key $P_A(x,y)$ comprises the steps of:
 - (i) entering a big odd integer p of size ≥ 160 bits;
 - (ii) determining whether p is a prime number;
 - (iii) going to next step if p is prime;
 - (iv) going to step (xv) if p is not prime;
 - (v) entering a small integer $a > 0$;
 - (vi) setting integer $b = 0$ & $x = 1$;

- (vii) determining whether $4a^3 + 27b^2 \text{ mod}(p) = \text{zero}$;
- (viii) going to next step if false;
- (ix) incrementing b by 1 if true and going to step (vii);
- (x) setting z ($=y^2$) $= x^3 + ax + b$;
- 5 (xi) determining whether $z (=y^2)$ is a perfect square;
- (xii) going to step (xxi) if z is not a perfect square;
- (xiii) setting B(x,y) equal to (x,y) if z is a perfect square;
- (xiv) selecting a large random integer r_1 ;
- (xv) setting $G(x,y) = (r_1 \cdot B(x,y)) \text{ mod}(p)$;
- 10 (xvi) selecting a large random integer n_A ;
- (xvii) setting $P_A(x,y) = (n_A \cdot G(x,y)) \text{ mod }(p)$;
- (xviii) return $P_A(x,y)$ as public key and n_A as private key;
- (xix) incrementing p by 2;
- (xx) returning to step (ii);
- 15 (xxi) incrementing x by 1;
- (xxii) determining whether $x > 900$;
- (xxiii) going to next step if true;
- (xxiv) going to step (x) if false;
- (xxv) incrementing b by 1;
- 20 (xxvi) setting x = 1;
- (xxvii) returning to step (vii).

9. A method of elliptic curve encryption as claimed in claims 1 to 8, wherein the said encryption of the said message MSG comprises the steps of:
- 25 (i) generating a large random integer K;
 - (ii) setting $P_{\text{mask}}(x,y) = k \cdot P_A(x,y) \text{ mod }(p)$;
 - (iii) setting $P_k(x,y) = k \cdot G(x,y) \text{ mod }(p)$;
 - (iv) accepting the message to be encrypted (MSG);
 - (v) converting the message into a point $P_c(x,y)$;

- (vi) generating a random point $P_m(x,y)$ on elliptic curve $E_p(a,b)$;
- (vii) setting $P_e(x,y) = (P_c(x,y) - P_m(x,y))$;
- (viii) setting $P_{mk}(x,y) = (P_m(x,y) + P_{mask}(x,y)) \text{ mod}(p)$;
- (ix) returning $P_k(x)$, $P_e(x,y)$ and $P_{mk}(x)$ as the result (cipher).

5 10. A method of elliptic curve encryption as claimed in claim 1 to 9, wherein the said decryption of the said ciphered text comprises the steps of:

- (i) getting cipher text ($P_k(x)$, $P_e(x,y)$, $P_{mk}(x)$);
- (ii) computing $P_k(y)$ from $P_k(x)$ using elliptic curve $E_p(a,b)$;
- 10 (iii) computing $P_{mk}(y)$ from $P_{mk}(x)$ using elliptic curve $E_p(a,b)$;
- (iv) computing $P_{ak}(x,y) = (n_A \cdot P_k(x,y)) \text{ mod}(p)$;
- (v) computing $P_m(x,y) = (P_{mk}(x,y) - P_{ak}(x,y)) \text{ mod}(p)$;
- (vi) computing $P_c(x,y) = P_m(x,y) + P_e(x,y)$;
- (vii) converting $P_c(x,y)$ into the input message MSG.

15 11. A method of elliptic curve encryption substantially as described and illustrated herein.

1/7

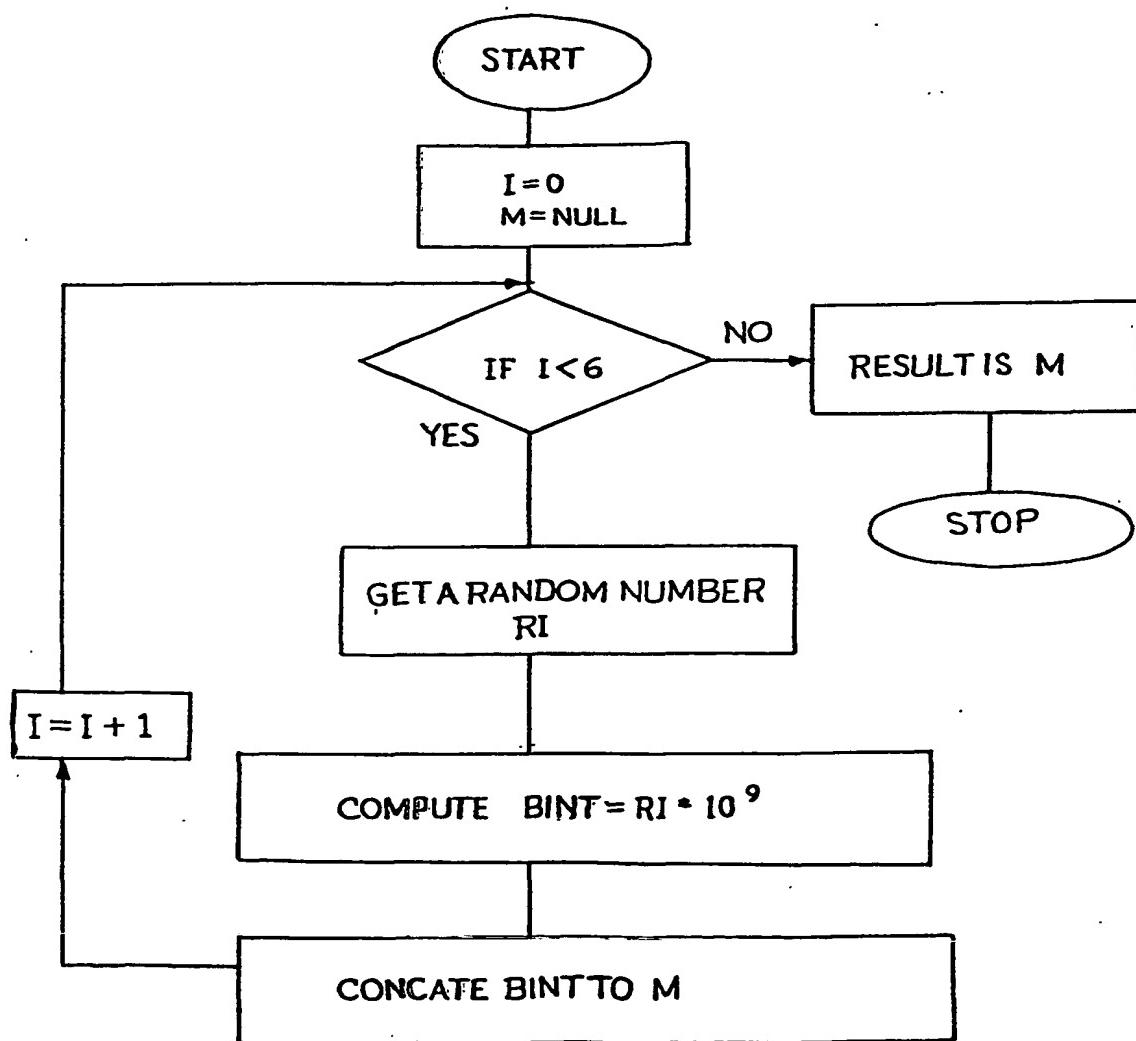


Fig. 1

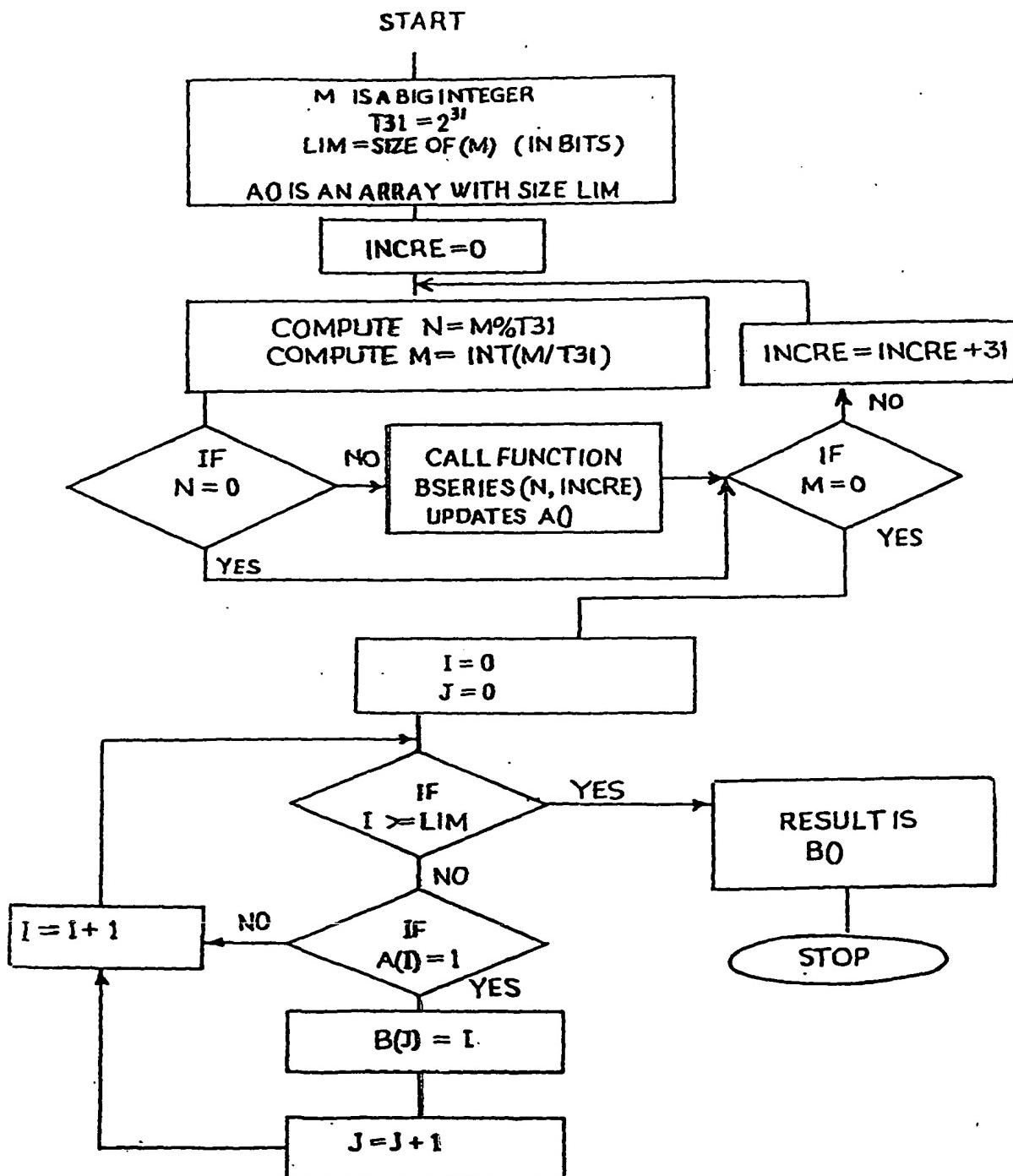


Fig. 2

3/7

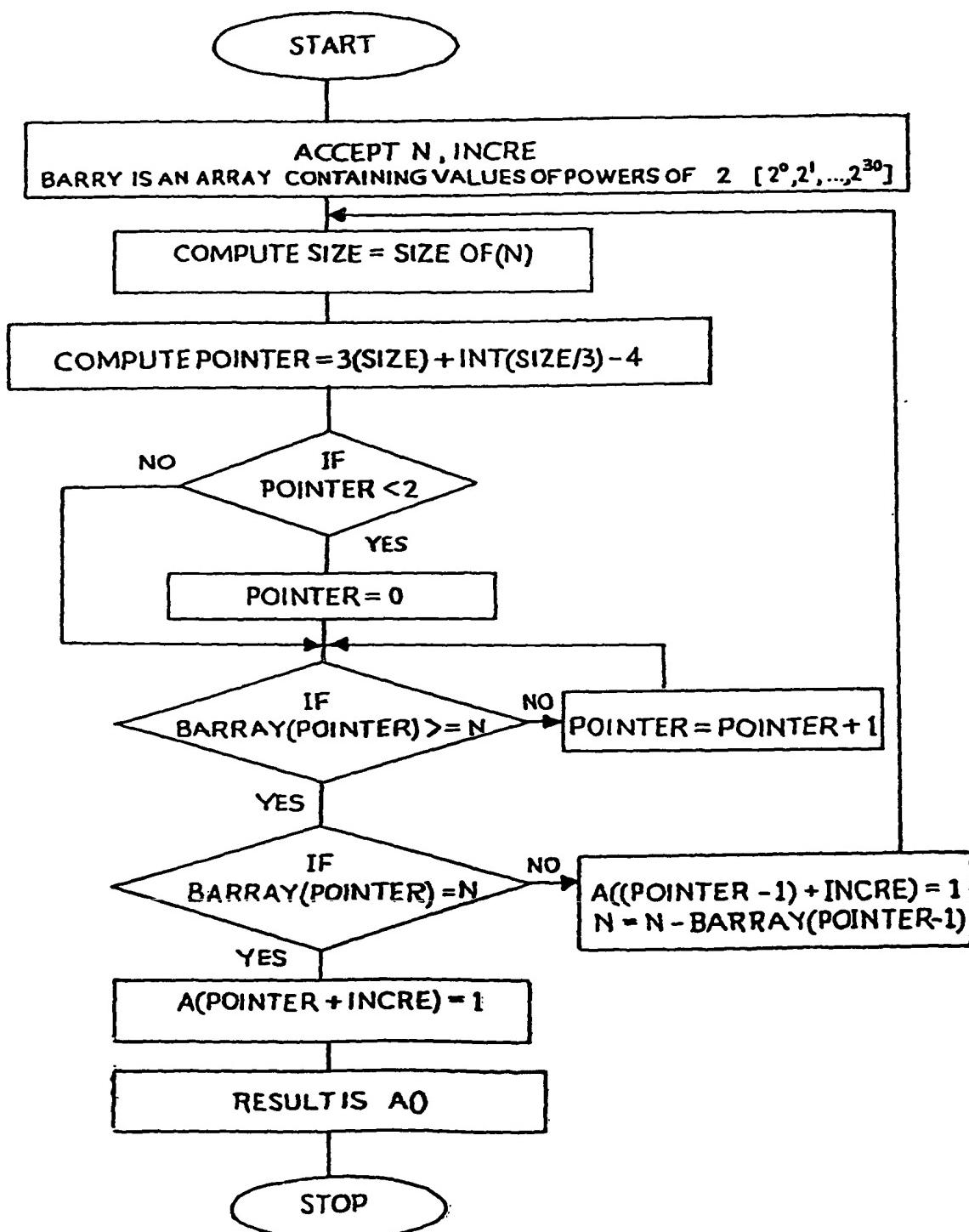


Fig. 3

4/7

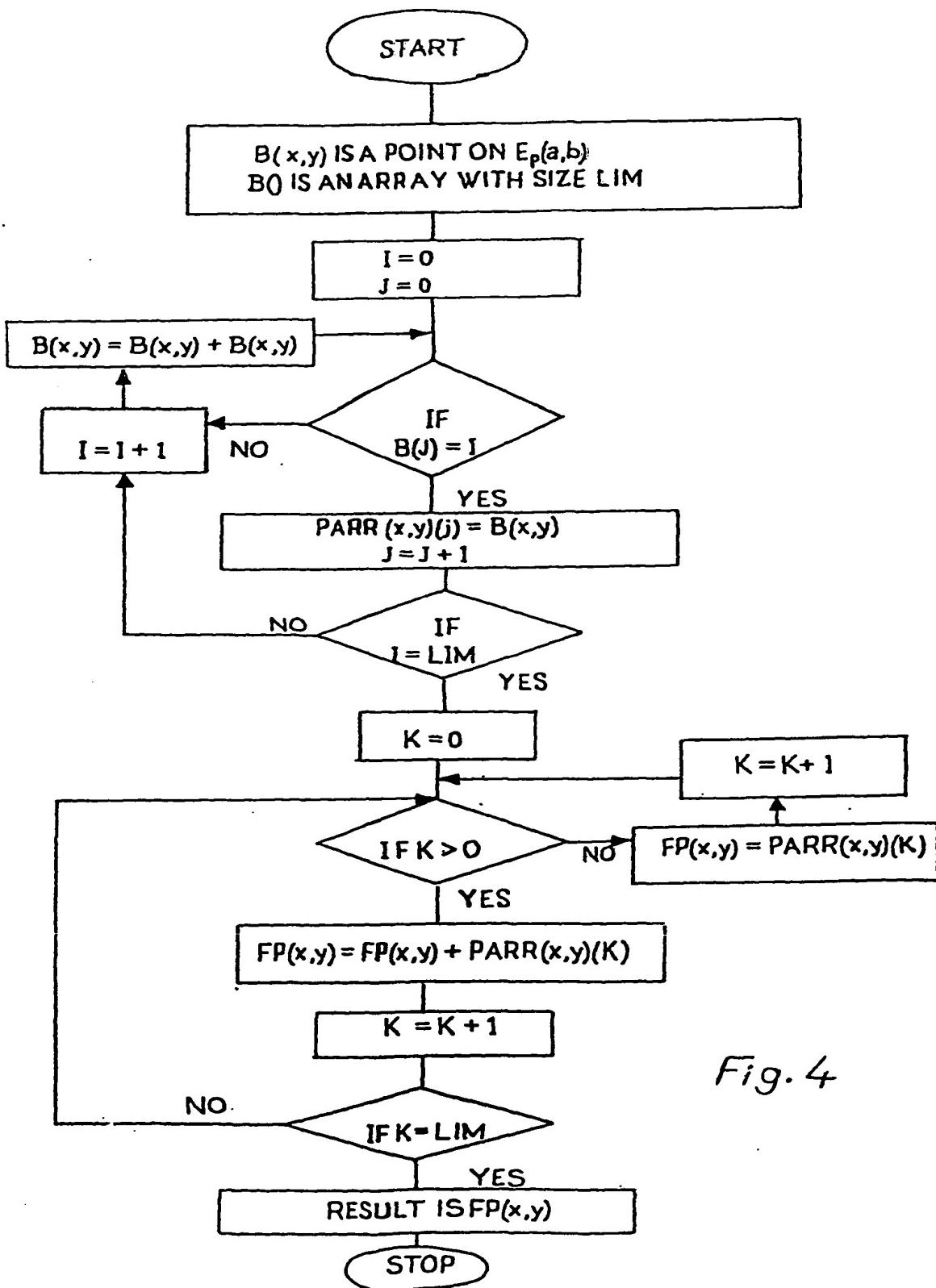


Fig. 4

5/7

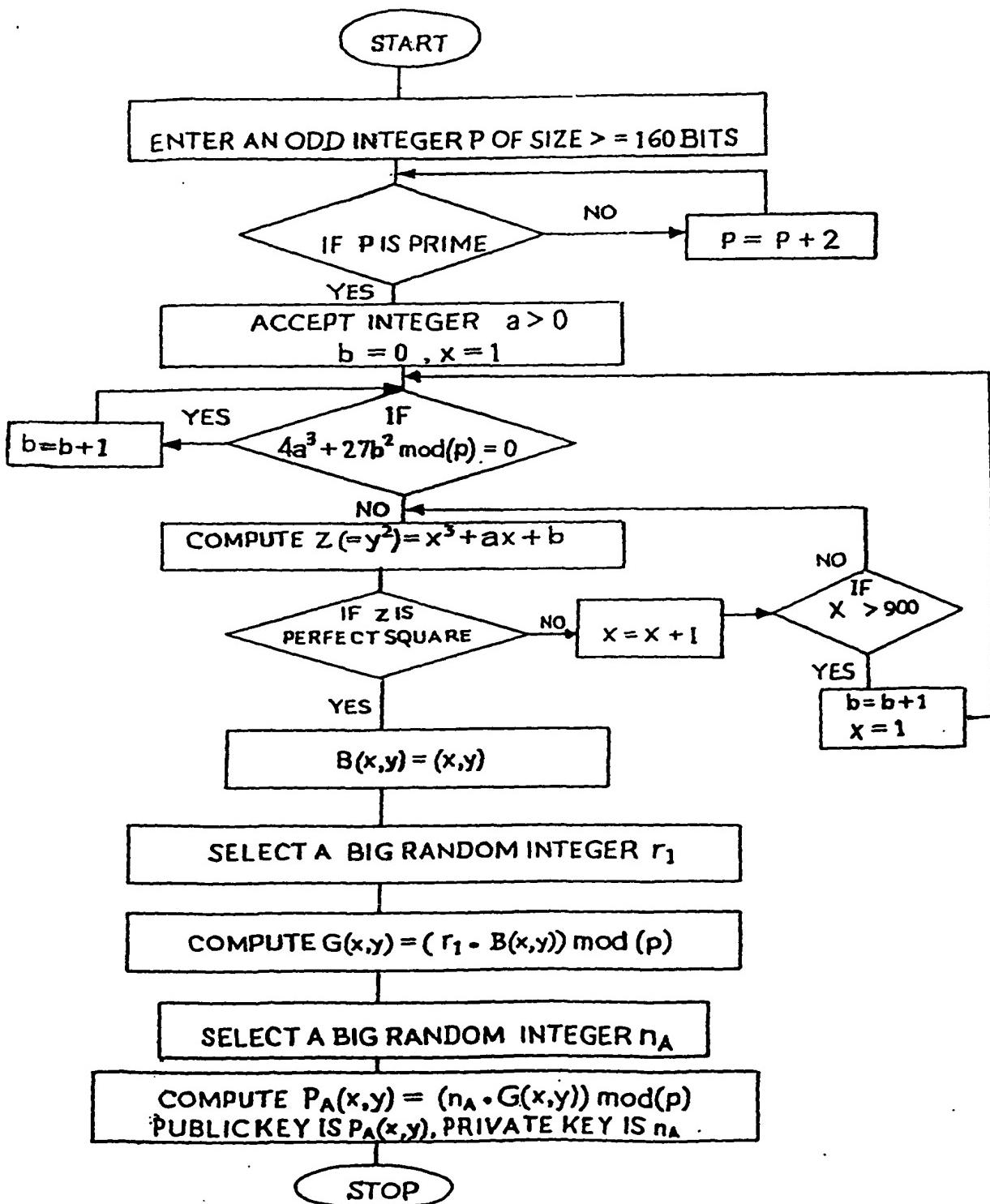


Fig. 5

6/7

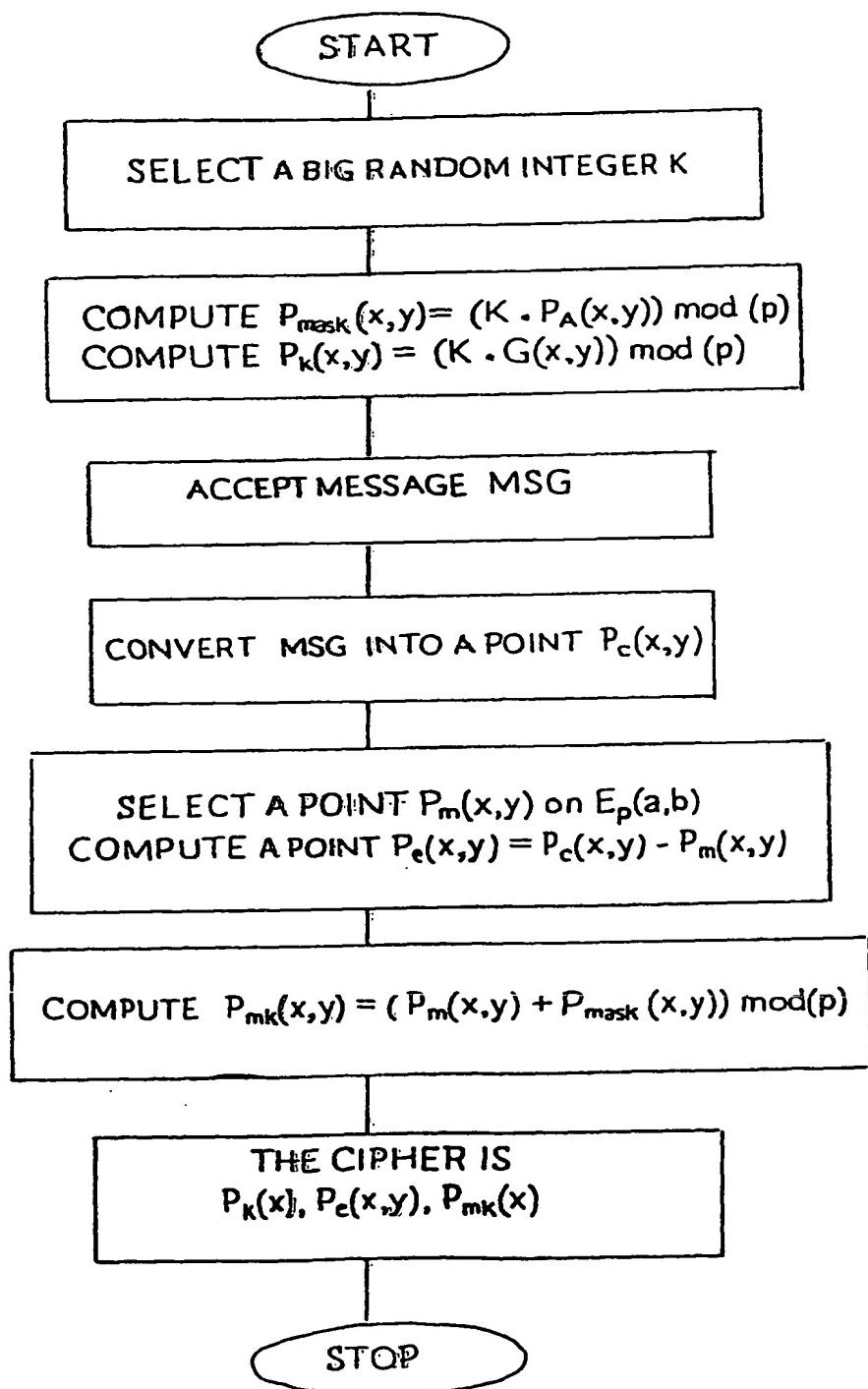


Fig. 6

7/7

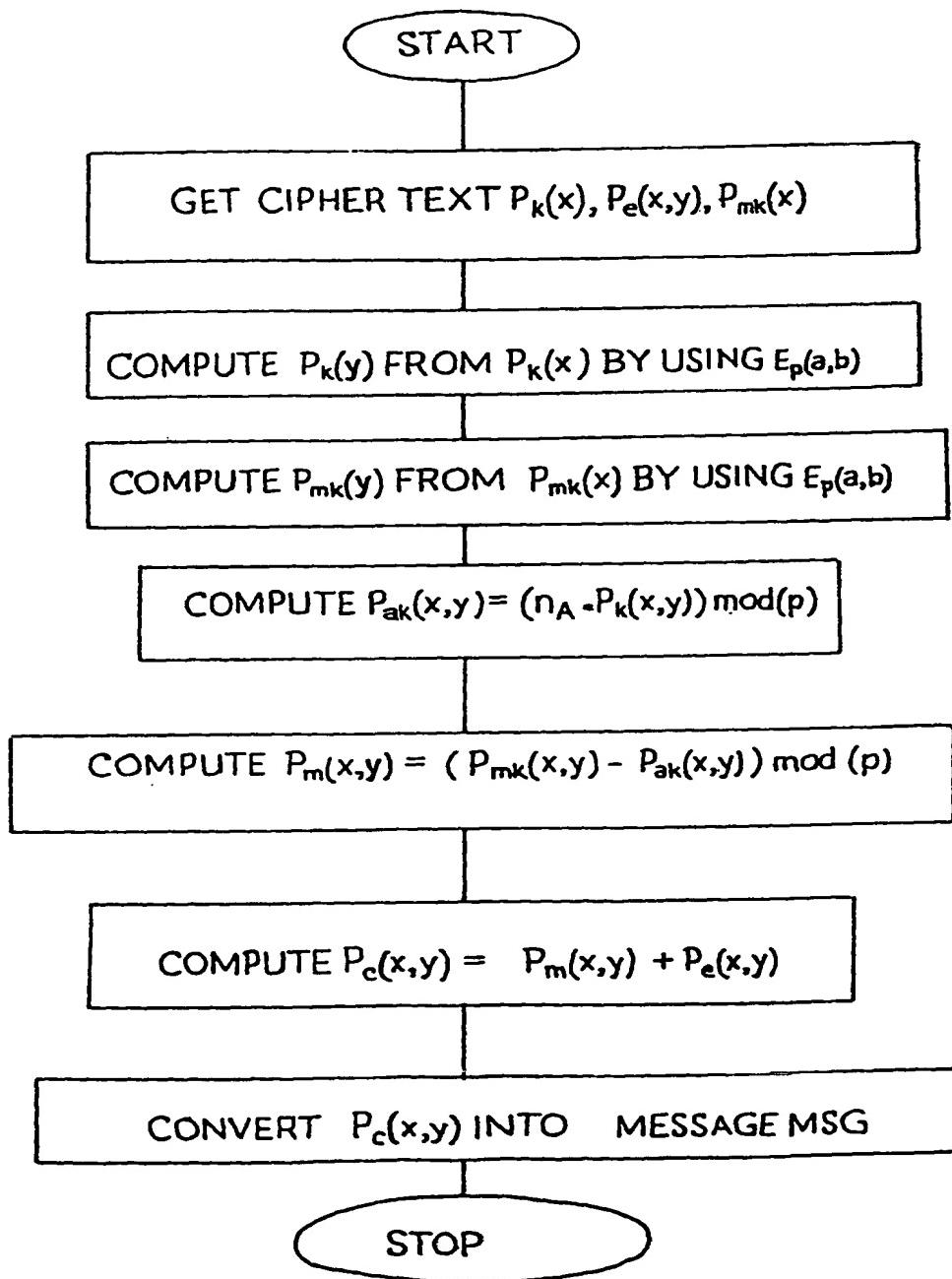


Fig. 7

INTERNATIONAL SEARCH REPORT

International Application No.
PCT/IN 03/00339-0

CLASSIFICATION OF SUBJECT MATTER

IPC⁷: G09C 1/00, H04L 9/30, 9/26

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC⁷: G09C, H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPODOC, WPI, PAJ, IEEE

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	WO 1994/015423 A1 (TELSTRA CORPORATION LIMITED) 7 July 1994 (07.07.1994) <i>fig 1; page 3, line 10 - page 6, line 20.</i>	1-10
A	EP 0503119 A1 (OMNISEC AG) 16 September 1992 (16.09.1992) <i>fig 2; abstract; page 14, line 53 - page 16, line 20.</i>	1-10
A	VANSTONE S. ET AL. "ELLIPTIC CURVE CRYPTOSYSTEMS USING CURVES OF SMOOTH ORDER OVER THE RING Zn" IEEE TRANSACTIONS ON INFORMATION THEORY, vol. 43, no. 4, July 1997. <i>pages 1231-1237.</i>	1-10

 Further documents are listed in the continuation of Box C. See patent family annex.

- * Special categories of cited documents:
- „A“ document defining the general state of the art which is not considered to be of particular relevance
 - „B“ earlier application or patent but published on or after the international filing date
 - „C“ document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
 - „D“ document referring to an oral disclosure, use, exhibition or other means
 - „E“ document published prior to the international filing date but later than the priority date claimed
 - „T“ later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
 - „X“ document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
 - „Y“ document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
 - „Z“ document member of the same patent family

Date of the actual completion of the international search 17 February 2004 (17.02.2004)	Date of mailing of the international search report 23 March 2004 (23.03.2004)
Name and mailing address of the ISA/AT Austrian Patent Office Dresdner Straße 87, A-1200 Vienna Facsimile No. 1/53424/535	Authorized officer WENNINGER W. Telephone No. 1/53424/325

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IN 03/00339-0

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
A			
EP A 3119		none	
	NO	A 790077	1979-07-12
	AT	A 16379	1981-01-15
	AT	B 363867B	1981-09-10
	EP	A 0003119	1979-07-25
	DE	A 2801034	1979-07-12
WO A 19940154 23		none	

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IN 03/00339-0

Box I Observations where certain claims were found unsearchable (Continuation of item 1 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. Claims Nos.: because they relate to subject matter not required to be searched by this Authority, namely:

2. Claims Nos.: 11 because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:
Claim 11 merely relates to the description and the figures and does not meet the demand for a distinguishing feature of the application. The international search report thus does not cover claim 11.

3. Claims Nos.: because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box II Observations where unity of invention is lacking (Continuation of item 2 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

1. As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.
2. As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.
3. As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

4. No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest

- The additional search fees were accompanied by the applicant's protest.
 No protest accompanied the payment of additional search fees.